SANSKRIT
SOFTWARE SYSTEMS

Module API

Programmer's Guide

Version: 1
Update: November, 2004

Contents

Introducion to Papirus

Papirus is a unique off-the-shelf software package providing total output management for AS/400. Papirus offers on-line connectivity between AS/400 applications and MS-Office applications running under the Windows operating system on a personal computer. By integrating the organizational management and data processing capabilities of the AS/400 with the advanced graphic capabilities of the popular Microsoft word processor, Papirus enables the organization to achieve a new level in document quality.

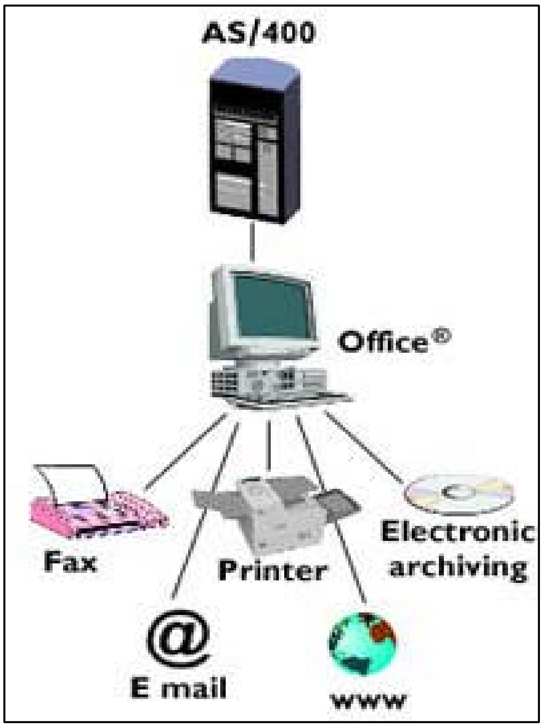


Figure 1. Papirus output management

In addition to the improvements in printing capabilities, Papirus enables the routing of the reshaped documents to any of the PC network printers, sends them out as a fax or e-mail, and enables connectivity to other PC applications such as a computerized archive or the Internet.

A single Papirus server provides print management and formatting services for the entire organization. Alternatively, several Papirus servers can be used in parallel to increase the print throughput.

After definition of the formatting parameters for company documents, Papirus works behind the scenes and is transparent to the user. The only variable elements are the company documents themselves, which are produced in a professional and attractive manner. This is done in the following way: an AS/400 application creates spool files, which are placed on the AS/400 output (print) queue. The regular Papirus AS/400 component (non API) captures these spools and attaches pre-defined formatting parameters (fonts, colors, tables, etc.) to create the final output documents. It then sends the spools with their formatting parameters from the AS/400 to the Papirus PC component. The PC component reformats the spool using MS-Word and sends the outcome document with its full graphic content to the desired output device.

Papirus can also convert spools into Excel spreadsheets. The Papirus Excel module is described in a separate user’s manual.

Introduction to AS/400 API

Papirus' AS/400 API allows you to produce MS-Word documents directly from AS/400 programs, bypassing the need for a spool file. Once connected to Papirus, your program can create formatted documents and distribute them through numerous devices. The creation of the formatted documents is done in two steps: creating a template document in Word and writing the AS/400 program.

The API has a number of unique advantages in both implementation and performance. First, it eliminates the need to characterize a spool file, which is usually a cumbersome process. Instead, the end user can create a Word document similar to his desired output document in structure and formatting. The IT department can then use that document as a template with which to produce the final document. Second, AS/400 resources which would otherwise be devoted to creating spool files, are now saved. Third, since there's no need to analyze spool data,

Papirus API is slightly faster than regular Papirus process and can even be faster than conventional AS/400 printing for certain print jobs.

Chapter 1 - The Template document

The template is a Word document which provides general structure for the output document. It defines settings such as page layout, fonts, colors and special formatting. Papirus will merge data from the AS400 into the template to create customized documents.

Creating the Template document

Open a new document in MS-Word and apply the desired page layout and margin settings on that document.

You may add headers and footers, auto text, Wordart and other functions from MS-Word (see figure 4 in the next page). The first line of the document, however, must remain empty.

Enter all the static data, which does not change from one document to another (the company's name, telephone number, titles etc.). This data may be formatted.

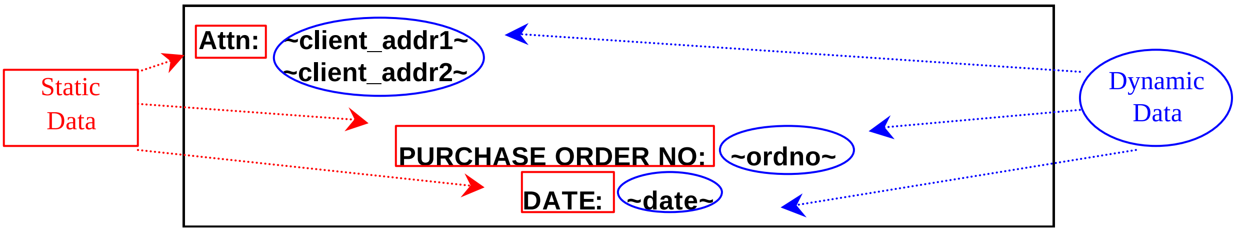
All the dynamic data, which changes in each report , should be entered to the template document as a variable name with the sign '~' before and after it. For example: ~name~.

Papirus will merge data from the AS400 into those variables. Not only can you replace the variables with text from the AS400, you may also replace them with images, documents and other PC objects.

The data transferred from the AS400 to the template will inherit the location of the variable and its formatting. If you want a header to be bold and underlined for example, you must apply this desired formatting on the variable: **~header~** .

While adding the variables into the template document, make sure the entire variable (name and ~ signs) is written in the same font and size.

Figure 2. Static and dynamic data in the template document



The final document might contain a table, which has an unknown number of raws.

In order to prepare a template for such a table, create a table with only two raws: one with headers and one with variables for the input.

Papirus will duplicate that one raw of variables for each line of data transferred from the AS400. Keep in mind that Papirus will duplicate the entire raw every time, even if just one column contains data.

Figure 3. template for a dynamic table

Catalog No.	Description	Model	Quantity	Unit Price	Total
~catalno~	~desc~	~draw~	~amnt~	~unitprice~	~total~

It is highly recommended using transparent tables to design the entire template document, even for non-tabular data, because it keeps the resulting document in order.

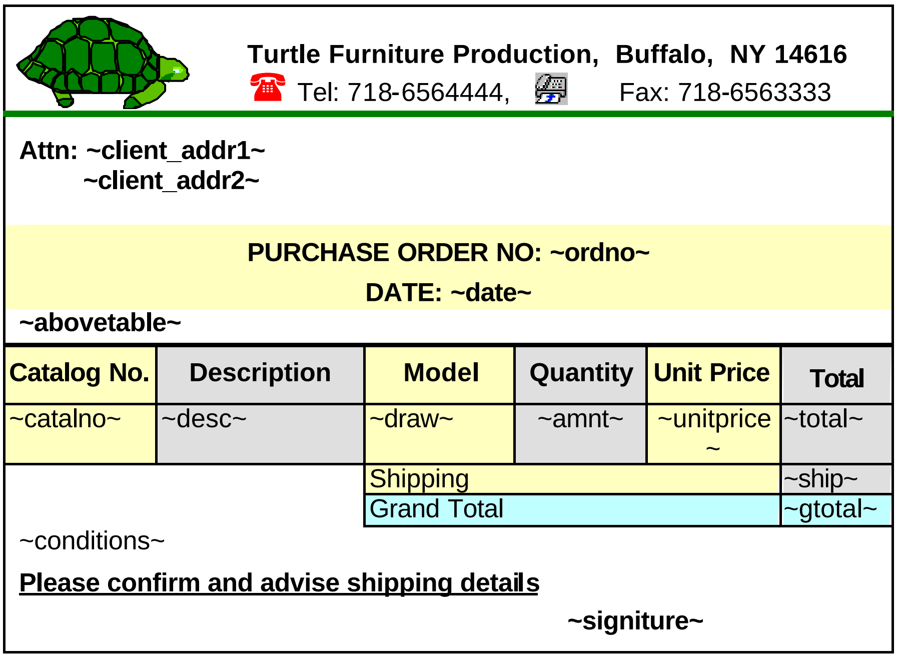
Saving the Template document

The template document must be saved in the forms directory on the Papirus PC server (the default is: C:\Program Files\Papirus\Forms) under any name up to 8 characters.

If you're using Word 97, the template documents should be saved in DOC format.

If you're using Word 2000, the template documents should be saved in DOT format.

Figure 4. Typical Template document



Turtle Furniture Production, Buffalo, NY 14616
☎ Tel: 718-6564444, 📠 Fax: 718-6563333

Attn: ~client_addr1~
~client_addr2~

PURCHASE ORDER NO: ~ordno~

DATE: ~date~

~abovetable~

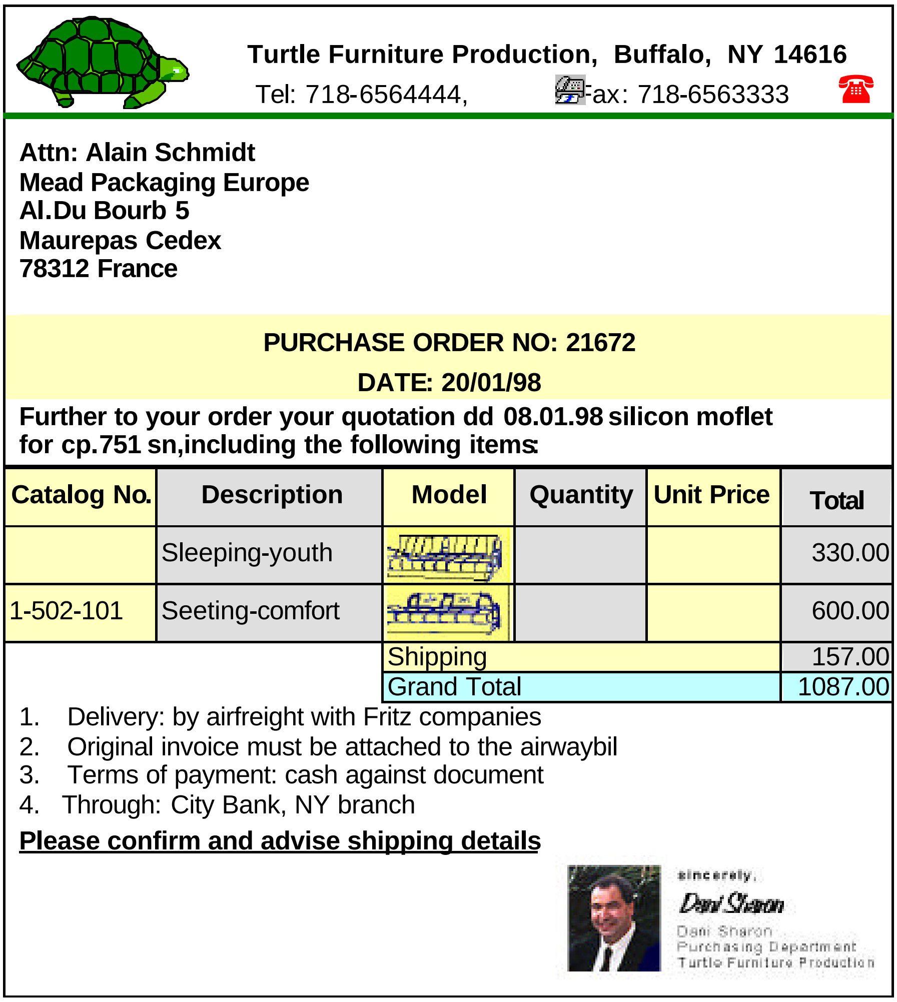
Catalog No.	Description	Model	Quantity	Unit Price	Total
~catalno~	~desc~	~draw~	~amnt~	~unitprice ~	~total~
		Shipping			~ship~
		Grand Total			~gtotal~

~conditions~

Please confirm and advise shipping details

~signiture~

Figure 5. Possible output document



Turtle Furniture Production, Buffalo, NY 14616
Tel: 718-6564444, 📠ax: 718-6563333 ☎

Attn: Alain Schmidt
Mead Packaging Europe
Al.Du Bourb 5
Maurepas Cedex
78312 France

PURCHASE ORDER NO: 21672

DATE: 20/01/98

Further to your order your quotation dd 08.01.98 silicon moflet for cp.751 sn,including the following items:

Catalog No.	Description	Model	Quantity	Unit Price	Total
	Sleeping-youth				330.00
1-502-101	Seeting-comfort				600.00
		Shipping			157.00
		Grand Total			1087.00

1. Delivery: by airfreight with Fritz companies
2. Original invoice must be attached to the airwaybil
3. Terms of payment: cash against document
4. Through: City Bank, NY branch

Please confirm and advise shipping details

sincerely,
Dani Sharon
Dani Sharon
Purchasing Department
Turtle Furniture Production

Package

A "package" is a feature of Papirus which enables you to dynamically merge several templates into a single output document.

In the previous section we learned how to create a template document, which holds the basic structures of the output document. Sometimes the desired output documents are very dynamic, for which it is almost impossible to create just one template. Other times you want every section in the output document to have a different appearance, designed by your choice.

In these cases you may create several different templates and Papirus will automatically combine them to produce one output document.

Using Package

Using the package option requires two things:

1) Create all the desired template documents separately and save them in the forms folder under different names.
2) Add the field “PACK” to the API program. This field marks to Papirus that more than one template document is being used. In this field you enter Start and End values to mark the begining and the end of each template. You can also use the field “DNBRK” to choose whether to merge the next page with a spacing line, with no spacing at all or with a page break between. For further details refer to page 10.

Once these two requirements are met, Papirus will automatically merge all the separate template documents into one output document.

Chapter 2 – AS/400 API

The AS/400 API creates a formatted Word document and then distribute it to desired output devices. In the program you define which data to transfer from the AS400 into the template, in which format to save the final document and where to direct it. A separate program must be written for each type of document (i.e. one for the packing lists, one for the invoices etc.).

Writing the API program

The API program consists of pre-defined fields and can be written in either RPG or Cobol.

It is made of two types of fields:

1) Document fields – fields defining the output document as a whole.
2) Variable fields – fields defining each separate variable in the template.

For further details about these fields please refer to page 9.

While defining which data to transfer to the template you must re-access Papirus for each variable individually. This is done by calling the program #PPAPI in the library PPIRUS.

The document fields will be transferred in the first call to Papirus. The variable fields along with the name of the template will be transferred from the first call on. The fields for each variable will be transferred in separate calls. The last call must contain the field DBNAME with the value *END**. This field marks the end of the data transfer and the beginning of the Word document creation.

Figure 6. Structure of the API program

Call No.	Parameters transferred
First	CALL #PPAPI Document fields Variable fields for the first variable along with the template name and number

Second - End	CALL #PPAPI Variable fields of each variable along with the template name and number
Last	CALL #PPAPI The field DBNAME='*END**'

Demo programs

RPG 3 programs: A demo program called PPAPIRPG may be found in the library PPIRUS under QRPGSRC. You may use all the sources with the name PPAPI* as well.

Cobol programs: A demo program called PPAPICBL may be found in the library PPIRUS under QCBLSRC. You may use all the sources with the name PPAPI* as well.

For further details about these demo programs refer to page 15.

API fields

A list of all the fields, from which to build the API program, appears in the following tables. Each field has different acceptable values and a required length. The size of all the fields should be 800 characters. The default value for all Y/N fields is N unless specified otherwise, so you may leave them blank in order to use the default option. Some of the fields are mandatory and must appear in every API program.

Document Fields

Use the document fields to connect your program to the relevant templates, to define the name and format of the output document, to set the printing parameters and all other options regarding the entire output document. Most of the fields should be transferred in the first call only. The fields WORDNM and LETNUM must be transferred in every call.

Field Name	From	To	Description	Acceptable Values	Remarks
WORDNM	1	8	Name of the template document	Any name up to 8 characters	Mandatory for every variable
CPYS	10	11	Number of copies to print.	01-99	
PRINT	12	12	Whether or not to print the output document.	N/Y	Default is Y
DTAQ	13	22	Name of the Dataq through which to transfer data from the AS400 to the PC. If kept blank, the one defined to the user or the default dataq is used.		For further details about Dataq and multiple servers refer to Papirus Word module user’s guide.
FORMS	23	32	Name of the folder on the PC containing the template documents.		Currently not in use.
PRT#DV	33	82	Name of the PC printer with which to print the		

Field Name	From	To	Description	Acceptable Values	Remarks
			output document. If kept blank, the one defined to the user or the default printer is used.		
PATH	83	13 2	The path on the PC in which to save the output document.		Available only with the field PCNAME.
PCNAME	133	14 0	Name of the output document created by Papirus.		If kept blank, an output document is not created.
SVTIFF	141	14 1	Save the output document in TIF format, not DOC format.	N/Y	
LETNUM	142	14 4	You may use one program to create several output documents. The number entered in this field separates the variables of each document. This number must remain constant through out the variables of each template.	If creating only one output document, enter 000	Mandatory for every variable
PACK	145	14 5	Use several templates to create one output document (package). Transfer the parameter S before the first field of the first template. Transfer the parameter E after the *END** of the last document.	S- Start E - End	
DNBRK	146	14 6	Ignore page break between the template documents in the package	1-merge next page with one spacing line. 2- merge	

Field Name	From	To	Description	Acceptable Values	Remarks
				next page with no spacing. Else – merge next page with a page break between.	
TXNAME	147	154	Name for the output document in TXT format.		
PRTY1	155	156	Set the priority in the dataq	01-99	
HTMLNM	157	186	Name for the output document in HTML format.		
REF1	187	216	Set a name or number with which to identify the document in Papirus' events log.		For further details about the events log refer to Papirus Word module user's guide.
R#ASM	490	499	Enter zeros in the field and they will later be replaced with an i.d. number given automatically to each document. This number is used to identify every document created by Papirus.	000000000	
DBNAME	500	506	Marks the end of the data transfer.	*END**	Mandatory in the last call.

Variable Fields

Use the variable fields to decide which data from the AS400 to transfer into each variable on the template and in which format.

Field Name	From	To	Description	Acceptable Values	Remarks
F#WORD	507	526	Variable name in the template document. For example: if you wrote in the template ~field1~, here you should write "field1" moved left.		Mandatory
F#LEN	527	529	Length of the data transferred into the variable		Calculated automatically if kept blank.
F#SOG	530	530	Type of the data transferred into the variable: alphabetical, numeric, date, binary or packed.	A – alphabetical S – numeric Y – date B – binary P - packed	Default is A
F#NKDA	531	532	Number of digits after the dot in numeric data. If kept blank, the number is considered 0		
F#K#AR	533	533	Editing codes for the different types of data. Only one editing code can be used.	For fields type A: M= Aligned center R = Aligned right L= Aligned left For fields type Y,B,S Z = suppress Zero	

Field Name	From	To	Description	Acceptable Values	Remarks
				K = Format number as x,xxx.xx	

Field Name	From	To	Description	Acceptable Values	Remarks
F#K#DT	534	534	Date editing code, for fields of type Y	1 – ddmmyy 2 – yymmdd 3 – mmddyy 4 – ddmmyyyy 5 – yyyymmdd 6 – mmddyyyy 7 – ddmmyyy 8 – yyymmdd 9 - mmddyyy	
F#ARH	535	684	Data to transfer into the template moved left		Mandatory

Papirus' Commands

The API program can include Papirus' commands as well. Use these commands to control the routing of the output document. In order to use Papirus' commands you must enter the name of the command into F#WORD and the value of the command into F#ARH. For example: to send the output document by email you must enter _EMAILTO_ into F#WORD and the actual email address into F#ARH.

Command	Description
AUTHOR	Set document author
DOCNAME	Set document printing label
DONTSEND	Cancel document transmission by email or fax
EMAILCC	Set email CC destination address, list with ";" separator is accepted
EMAILCCP	Set email CC destination address for the entire package
EMAILDOCN	Set email attachment name
EMAILFORMAT	Set email attachment format (D – doc, H – html, T – txt, G – gif, P – pdf)
_EMAILFROM _	Set email sender address*
EMAILFROMNAME	Set email sender full name*
EMAILTEXT	Set email message text
EMAILTO	Set email TO destination address, list with ";" separator is accepted
EMAILTOP	Set email TO destination address for the entire package
FAXNUMBER	Set fax destination number, list with ";" separator is accepted
FAXNUMP	Set fax destination number for the entire package
IMGCOLOR	Set output document IMAGE to b/w or color**
IMGRES	Set output document IMAGE resolution**
IMGSCALE	Set output document IMAGE scale factor**
IMGTYPE	Set output document IMAGE type**
MACRO	Run word macro

Command	Description
_MACRO_DOTFILE_	Macro file location
PASSWORD	Set email attachment protection password, by default the password is 777.
PATH	Set search path for objects in the PC
PDFNAME	Set document PDF output name
PRINTAFTER	Print DOC after main print
PRINTBEFORE	Print DOC before main print
REMARK	Set fax remark in AS/400 events log
RUN	Activate AS/400 command on document completion
_SMS_NUMBER_	Set SMS destination number
_SMS_MSG_	Set SMS message
SUBJECT	Set fax & email subject
TEST	Open the output document in MS Word Viewer automatically
TIFFNAME	Set output document IMAGE name
_TMPL_DOCNAME_	Override the template document name
_TMPL_PAGEMODE_	Override template page mode

* Papirus can send email using Outlook or directly through SMTP protocol. These options are available only under SMTP configuration.

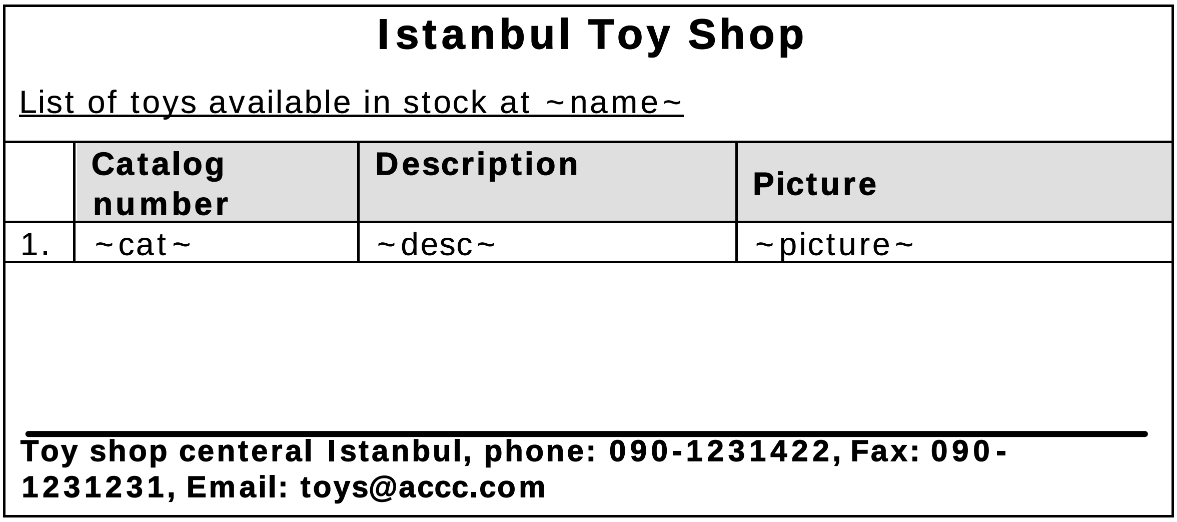
** Refers to image formats such as JPG, TIFF, GIF, BMP etc.

The full list of Papirus' commands can also be found in the spool interface in the Field ruler window (refer to Papirus Word module user's guide).

Appendix A – Sample API program

The program PPAPIRPG in library PPIRUS is an example program which creates a Word document using Papirus' API module. The program uses a template document called sample1.doc located in the folder Papirus\Sample on the Papirus PC server.

Figure 6. Template document sample1.doc



Istanbul Toy Shop

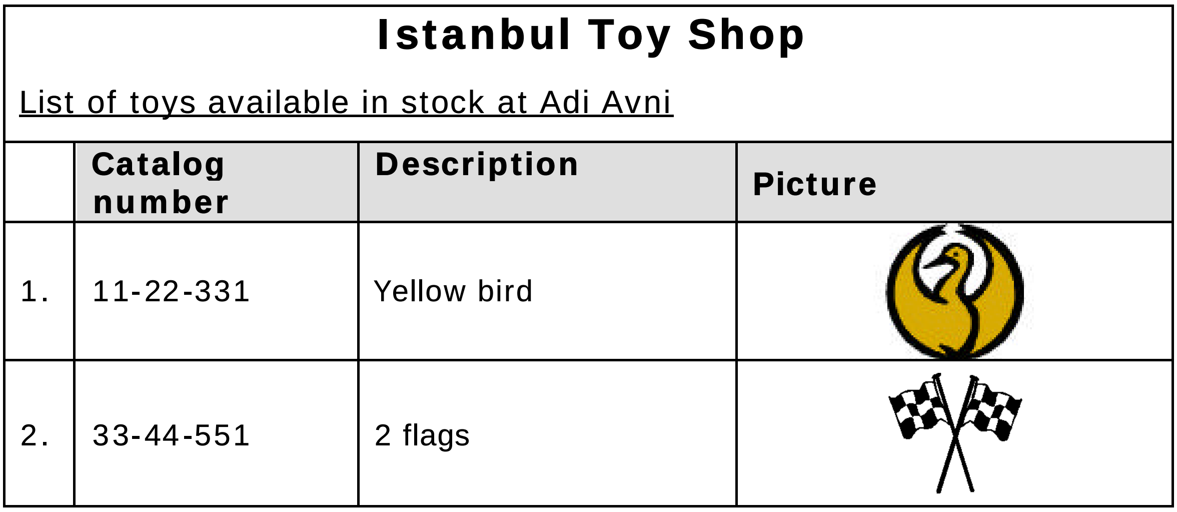
List of toys available in stock at ~name~

	Catalog number	**Description**	**Picture**
1.	~cat~	~desc~	~picture~

Toy shop centeral Istanbul, phone: 090-1231422, Fax: 090-1231231, Email: toys@accc.com

This template contains four variables: ~name~ , ~cat~ , ~desc~ and ~picture~. The API program replaces the variable ~name~ with the text 'Adi Avni' and sends four sets (lines) of data into the three other variables, creating a Word table.

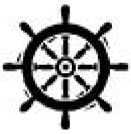
Figure 7. Output document created from template sample1.doc



Istanbul Toy Shop

List of toys available in stock at Adi Avni

	Catalog number	**Description**	**Picture**
1.	11-22-331	Yellow bird	
2.	33-44-551	2 flags	

3.	88-99-987	Wheel	
4.	25-12-212	Scales	
Toy shop centeral Istanbul, phone: 090-1231422, Fax: 090-1231231, Email: toys@accc.com			

Stracture of the sample API Program

```
H***************************************************************

E*   4 Catalog numbers

E                  CT      1   4  9

E*   4 Descriptions

E                  DS      1   4 25

E*   4 Names of BMP files

H*  and sends 4 sets (lines) of the 3 other fileds, which create together a WORD table.

H***************************************************************

E*   4 Catalog numbers

E                  CT      1   4  9

E*   4 Descriptions

E                  DS      1   4 25

E*   4 Names of BMP files

E                  PC      1   4 15

I/COPY PPIRUS/QRPGSRC,PPAPII

C*************************************

C*  Papirus First field of template   *

C*************************************

C*   This section was copied from PPIRUS/PPAPIFS
```

```
I/COPY PPIRUS/QRPGSRC,PPAPII
C************************************
C*  Papirus First field of template  *
C************************************
C*   This section was copied from PPIRUS/PPAPIFS
C*   Parameter WORDNM change from DOCNAME to SAMPLE1
C*   Parameter F#WORD change from FLDNAME to NAME
C*   Parameter F#LEN  change from 5       to 25
C*   Parameter F#ARH  change from XXXXX   to ADI AVNI
C                    CLEARKELET
C*  Doc name
C                    MOVEL'SAMPLE1' WORDNM
C*  Print doc Y/N Defult = Y
C                    MOVE 'Y'       PRINT
C*   Parameter F#WORD change from FLDNAME to NAME
C*   Parameter F#LEN  change from 5       to 25
C*   Parameter F#ARH  change from XXXXX   to ADI AVNI
C                    CLEARKELET
C*  Doc name
C                    MOVEL'SAMPLE1' WORDNM
C*  Print doc Y/N Defult = Y
C                    MOVE 'Y'       PRINT
C*  Word field name
C                    MOVEL'NAME'    F#WORD
C*  Field length
C                    Z-ADD25        F#LEN
C*  Field type
C                    MOVEL'A'       F#SOG
C*  Field value
C                    MOVEL'ADI AVNI'F#ARH
```

```
C*  Papirus Api program
C                 MOVEL'NAME'    F#WORD
C*  Field length
C                 Z-ADD25        F#LEN
C*  Field type
C                 MOVEL'A'       F#SOG
C*  Field value
C                 MOVEL'ADI AVNI'F#ARH
C*  Papirus Api program
C                 MOVEL'PPIRUS/' PPPGMN 13
C                 MOVE '#PPAPI'  PPPGMN
C*  Papirus data library
C                 CALL PPPGMN
C                 PARM           KELET
C******************************************************
C*   Loop 4 times for 4 lines of the table
C         1       DO   4         I      20
C                 MOVE '#PPAPI'  PPPGMN
C*  Papirus data library
C                 CALL PPPGMN
C                 PARM           KELET
C******************************************************
C*   Loop 4 times for 4 lines of the table
C         1       DO   4         I      20
C*   This section was copied from PPIRUS/PPAPIF
C*   Parameter WORDNM change from DOCNAME to SAMPLE1
C*   Parameter F#WORD change from FLDNAME to CAT
C*   Parameter F#LEN  change from 5       to 9
C*   Parameter F#ARH  change from XXXXX   to CT,I
C*  Clear parameters
```

```
C                  CLEARKELET
C*  Doc name
C*   Parameter WORDNM change from DOCNAME to SAMPLE1
C*   Parameter F#WORD change from FLDNAME to CAT
C*   Parameter F#LEN  change from 5       to 9
C*   Parameter F#ARH  change from XXXXX   to CT,I
C*  Clear parameters
C                  CLEARKELET
C*  Doc name
C                  MOVEL'SAMPLE1' WORDNM
C*  Word field name
C                  MOVEL'CAT'    F#WORD
C*  Field length
C                  Z-ADD9        F#LEN
C*  Field type
C                  MOVEL'A'      F#SOG
C*  Field value
C                  MOVELCT,I     F#ARH
C*  Word field name
C                  MOVEL'CAT'    F#WORD
C*  Field length
C                  Z-ADD9        F#LEN
C*  Field type
C                  MOVEL'A'      F#SOG
C*  Field value
C                  MOVELCT,I     F#ARH
C                  CALL PPPGMN
C                  PARM          KELET
C*   Parameter WORDNM change from DOCNAME to SAMPLE1
C*   Parameter F#WORD change from FLDNAME to DESC
```

```
C*   Parameter F#LEN  change from 5       to 25
C*   Parameter F#ARH  change from XXXXX   to DS,I
C*  Clear parameters
C                  CLEARKELET
C                  PARM           KELET
C*   Parameter WORDNM change from DOCNAME to SAMPLE1
C*   Parameter F#WORD change from FLDNAME to DESC
C*   Parameter F#LEN  change from 5       to 25
C*   Parameter F#ARH  change from XXXXX   to DS,I
C*  Clear parameters
C                  CLEARKELET
C*  Doc name
C                  MOVEL'SAMPLE1' WORDNM
C*  Word field name
C                  MOVEL'DESC'    F#WORD
C*  Field length
C                  Z-ADD25        F#LEN
C*  Field type
C                  MOVEL'A'       F#SOG
C*  Field value
C                  MOVEL'SAMPLE1' WORDNM
C*  Word field name
C                  MOVEL'DESC'    F#WORD
C*  Field length
C                  Z-ADD25        F#LEN
C*  Field type
C                  MOVEL'A'       F#SOG
C*  Field value
C                  MOVELDS,I      F#ARH
C                  CALL PPPGMN
```

```
C                    PARM          KELET
C*   Parameter WORDNM change from DOCNAME to SAMPLE1
C*   Parameter F#WORD change from FLDNAME to PICTURE
C*   Parameter F#LEN  change from 5       to 15
C*   Parameter F#ARH  change from XXXXX   to PC,I
C*  Clear parameters C                  MOVEL'SAMPLE1' WORDNM
C*  Word field name
C                    MOVEL'DESC'    F#WORD
C*  Field length
C                    Z-ADD25        F#LEN
C*  Field type
C                    MOVEL'A'       F#SOG
C*  Field value
C                    MOVELDS,I      F#ARH
C                    CALL PPPGMN
C                    PARM          KELET
C*   Parameter WORDNM change from DOCNAME to SAMPLE1
C*   Parameter F#WORD change from FLDNAME to PICTURE
C*   Parameter F#LEN  change from 5       to 15
C*   Parameter F#ARH  change from XXXXX   to PC,I
C*  Clear parameters
C                    CALL PPPGMN
C                    PARM          KELET
C*   Parameter WORDNM change from DOCNAME to SAMPLE1
C*   Parameter F#WORD change from FLDNAME to PICTURE
C*   Parameter F#LEN  change from 5       to 15
C*   Parameter F#ARH  change from XXXXX   to PC,I
C*  Clear parameters
C                    CLEARKELET
C*  Doc name
```

```
C                  MOVEL'SAMPLE1' WORDNM
C*  Word field name
C                  MOVEL'PICTURE' F#WORD
C*  Field length
C                  Z-ADD15        F#LEN
C*  Field type
C                  MOVEL'A'       F#SOG
C*  Doc name
C                  MOVEL'SAMPLE1' WORDNM
C*  Word field name
C                  MOVEL'PICTURE' F#WORD
C*  Field length
C                  Z-ADD15        F#LEN
C*  Field type
C                  MOVEL'A'       F#SOG
C*  Field value
C                  MOVELPC,I      F#ARH
C                  CALL PPPGMN
C                  PARM           KELET
C                  ENDDO
C************************************************************
*
C*   Using Papirus command called _TEST_
C                  PARM           KELET
C                  ENDDO
C************************************************************
*
C*   Using Papirus command called _TEST_
C*   Parameter WORDNM change from DOCNAME to SAMPLAE1
C*   Parameter F#WORD change from FLDNAME to _TEST_
```

```
C*   Parameter F#ARH  change from XXXXX   to *YES

C*  Clear parameters

C                    CLEARKELET

C*  Doc name

C                    MOVEL'SAMPLE1' WORDNM

C*  Word field name

 C*   Parameter WORDNM change from DOCNAME to SAMPLAE1

C*   Parameter F#WORD change from FLDNAME to _TEST_

C*   Parameter F#ARH  change from XXXXX   to *YES

C*  Clear parameters

C                    CLEARKELET

C*  Doc name

C                    MOVEL'SAMPLE1' WORDNM

C*  Word field name

C                    MOVEL'_TEST_'  F#WORD

C*  Field value

C                    MOVEL'*YES'    F#ARH

C                    CALL PPPGMN

C                    PARM           KELET

C************************************************************
*

C*   This section was copied from PPIRUS/PPAPIFE

C*  Field value

C                    MOVEL'*YES'    F#ARH

C                    CALL PPPGMN

C                    PARM           KELET

C************************************************************
*

C*   This section was copied from PPIRUS/PPAPIFE

C*   Parameter WORDNM change from DOCNAME to SAMPLE1
```

```
C*  Clear parameters
C                    CLEARKELET
C*  Doc name
C                    MOVEL'SAMPLE1' WORDNM
C*  End indication
C                    MOVEL'*END'    DBNAME
C*   Parameter WORDNM change from DOCNAME to SAMPLE1
C*  Clear parameters
     C                    CLEARKELET
     C*  Doc name
     C                    MOVEL'SAMPLE1' WORDNM
     C*  End indication
     C                    MOVEL'*END'    DBNAME
     C                    CALL PPPGMN
     C                    PARM         KELET
C                    SETON                    LR
 **
 11-22-331
 33-44-551
 88-99-987
 25-12-212
 C                    PARM         KELET
C                    SETON                    LR
**
11-22-331
33-44-551
88-99-987
25-12-212
**
Yellow bird
```

```
2 flags
Wheel
Scales
**
@BMP:PICT1@
@BMP:PICT2@
@BMP:PICT3@
 Yellow bird
2 flags
Wheel
Scales
**
@BMP:PICT1@
@BMP:PICT2@
@BMP:PICT3@
@BMP:PICT4@
****************** End of data
****************************************
```